ФПМИ БГУ

Вычислительные методы алгебры
Лаборатоная работа 7

# Нахождение собственных значений и собственных векторов методом Крылова

Подготовил:
Ткачук Павел
2 курс 1 группа

Преподаватель:
Будник Анатолий Михайлович

20 декабря 2016 г.

# 1 Постановка задачи

Входные данные:

$$A = \begin{bmatrix} 0.6444 & 0.0000 & -0.1683 & 0.1184 & 0.1973 \\ -0.0395 & 0.4208 & 0.0000 & -0.0802 & 0.0263 \\ 0.0132 & -0.1184 & 0.7627 & 0.0145 & 0.0460 \\ 0.0395 & 0.0000 & -0.0960 & 0.7627 & 0.0000 \\ 0.0263 & -0.0395 & 0.1907 & -0.0158 & 0.5523 \end{bmatrix}$$

Для того чтобы все собственные значения были действительными числами, домножим слева исходную матрицу на ей транспонированную:

$$A = A_{\text{исх.}} A^T_{\text{исх.}} = \begin{bmatrix} 0.4965221 & -0.02976049 & -0.10906373 & 0.13191428 & 0.09195098 \\ -0.02976049 & 0.18575662 & -0.05029722 & -0.06272879 & -0.0018678 \\ -0.10906373 & -0.05029722 & 0.59823034 & -0.06163865 & 0.17564755 \\ 0.13191428 & -0.06272879 & -0.06163865 & 0.59248754 & -0.02931901 \\ 0.09195098 & -0.0018678 & 0.17564755 & -0.02931901 & 0.34390336 \end{bmatrix}$$

Задача:

1. Для данной матрицы $A$ вычислить коэфициенты $p_1, p_2, \ldots, p_n$ собственного многочлена

$$P(A) = \lambda^n - p_1\lambda^{n-1} - p_2\lambda^{n-2} - \ldots - p_{n-1}\lambda - p_n$$

   (проверить совпадает ли $p_1 = Sp(A)$ и $p_n = \det(A)$)

2. По полученному характеристическому многочлену вычислить максимальное собственное значение $\lambda_{max}$ матрицы

3. Вычислить собственный вектор $\vec{v}$ соответствующий данному значению

4. Найти невязку $\vec{r} = A\vec{v} - \lambda_{max}\vec{v}$

# 2 Алгоритм

1. Вычисляем вектора $c^{(i)},\ i = \overline{0, n}$

$$c^{(0)} = (1, 0, 0, \ldots, 0),$$
$$c^{(i)} = A^i c^{(0)},\ i = 1, \ldots, n.$$

2. Решаем систему

$$\begin{cases} q_1 c_1^{(n-1)} + \ldots + q_n c_1^{(0)} = c_1^{(n)}, \\ \vdots \\ q_1 c_n^{(n-1)} + \ldots + q_n c_n^{(0)} = c_n^{(n)}. \end{cases}$$

   Полученное решение $(q_1, q_2, \ldots, q_n)$ представляет собой коэфициенты $(p_1, p_2, \ldots, p_n)$ характеристического многочлена исходной матрицы

3. Находим $\lambda_{max}$ - макисмальный корень характеристического уравнения

$$\lambda^n - p_1\lambda^{n-1} - p_2\lambda^{n-2} - \ldots - p_{n-1}\lambda - p_n = 0$$

4. По полученному $\lambda_{max}$ вычисляем собственный вектор, соотвествующий данному значению

$$\vec{v} = \beta_1 c^{(n-1)} + \ldots + \beta_n c^{(0)}, \text{ где}$$
$$\beta_1 = 1,$$
$$\beta_i = \lambda_{max}\beta_{i-1} - q_{i-1},\ i = 2, \ldots, n.$$

# 3 Результаты и вывод

## 3.1 Входные данные

0.6444 0.0000 -0.1683 0.1184 0.1973
-0.0395 0.4208 0.0000 -0.0802 0.0263
0.0132 -0.1184 0.7627 0.0145 0.0460
0.0395 0.0000 -0.0960 0.7627 0.0000
0.0263 -0.0395 0.1907 -0.0158 0.5523

## 3.2 Выходные данные

```
Ищем собственные вектор матрицы
Исходная матрица:
[[ 0.4965221  -0.02976049 -0.10906373  0.13191428  0.09195098]
 [-0.02976049  0.18575662 -0.05029722 -0.06272879 -0.0018678 ]
 [-0.10906373 -0.05029722  0.59823034 -0.06163865  0.17564755]
 [ 0.13191428 -0.06272879 -0.06163865  0.59248754 -0.02931901]
 [ 0.09195098 -0.0018678   0.17564755 -0.02931901  0.34390336]]
Коэфициенты характеристического многочлена
 [ 2.21689996 -1.82259145  0.68304846 -0.11469321  0.0069546 ]
q_1-Sp(A) =  5.77315972805e-15 q_n-det(A) =  1.552577511e-16
Собственное значение:  0.780834944861
Собственный вектор:  [ 0.00382651 -0.00022833 -0.00553064  0.00483712 -0.00174166]
Невязка:  [  1.56125113e-17  -5.74627151e-18  -3.64291930e-17   2.47198095e-17
   6.28837260e-18]
Норма невязки:  4.74812474615e-17
```

## 3.3 Вывод

Данный метод относится к точным методам и позволяет решать проблему собственных значений. С его помощью можно находить точное решение данной проблемы, единственные погрешности которые возникают, возникают за счёт неточности машинных вычислений.

# 4 Листинг кода

```
import numpy as np
import numpy.linalg as linalg


# Поиск собственных векторов методом Крылова


def eigen(A):
    size = len(A)
    C = np.eye(size)
    for i in range(1, size):
        C[:, i] = np.dot(A, C[:, i - 1])
    c = np.dot(A, C[:, size - 1])
    C = C[:, ::-1]
    q = linalg.solve(C, c)
    print("Коэфициенты характеристического многочлена\n", q)
    print("q_1-Sp(A) = ", abs(A.trace() - q[0]), "q_n-det(A) = " ,abs(linalg.det(A) - q[size - 1]))
    p = [1] + list(q * -1) # Получили коэфициенты собственного многочлена
    eigvals = np.roots(p) # Находим его корни

    beta = np.zeros(size) # Вычисляем собственный вектор
    beta[0] = 1
    for i in range(1, size):
        beta[i] = beta[i - 1] * eigvals[0] + p[i]
    return sum([beta[i] * C[:, i] for i in range(size)]), eigvals[0]


file = open("matrix", "r")  # Чтение файла
A, b = [], []
for line in file:
    A.append([float(el) for el in line.split()[:-1]])
    b.append(float(line.split().pop()))
A = np.array(A)
A = np.dot(A, A.transpose())
b = np.array(b)
print("Ищем собственные вектор матрицы")
print("Исходная матрица:")
print(A)
ans = eigen(A)
print("Собственное значение: ", ans[1])
print("Собственный вектор: ", ans[0])
print("Невязка: ", np.dot(A, ans[0]) - ans[0] * ans[1])
print("Норма невязки: ", linalg.norm(np.dot(A, ans[0]) - ans[0] * ans[1]))
```